



# Introdução à Criptografia

**Prof. Dr. Iaçanã Ianiski Weber**

*Confiabilidade e Segurança de Software*

98G08-4

*Agradecimentos especiais ao Prof. Avelino Zorzo e aos Autores Christof Paar e Jan Pelzl pelo material.*

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Objetivos Clássicos da Segurança da Informação

- ① **Privacidade:** sem vazar dados confidenciais
- ② **Autenticação:** sem se passar por outro
- ③ **Integridade:** sem alteração
- ④ **Não-repúdio:** não ser capaz de negar

*Esses quatro pilares formam a base da segurança em sistemas computacionais.*

# Leituras e Informações Adicionais

Paar, Christof, and Jan Pelzl. **Understanding Cryptography: A Textbook for Students and Practitioners**. Springer, 2010.

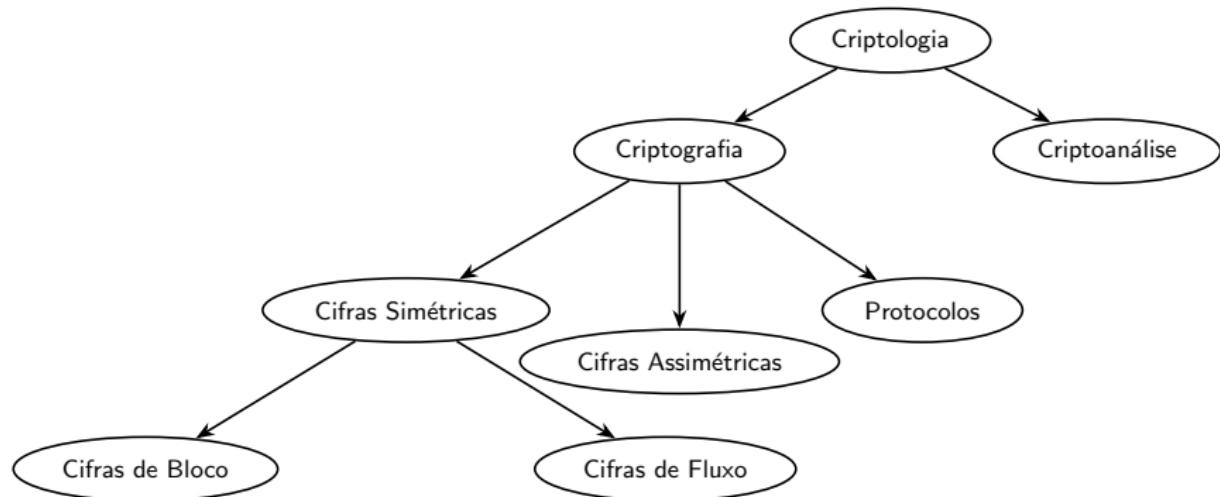
## Complementar ao Understanding Cryptography:

- A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, outubro de 1996.
- H. v. Tilborg (ed.), *Encyclopedia of Cryptography and Security*, Springer, 2005.

## História da Criptografia (ótima leitura para a noite):

- S. Singh, *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography*, Anchor, 2000.
- D. Kahn, *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet*. 2<sup>a</sup> edição, Scribner, 1996.

# Classificação da Área da Criptologia



# Alguns Fatos Básicos

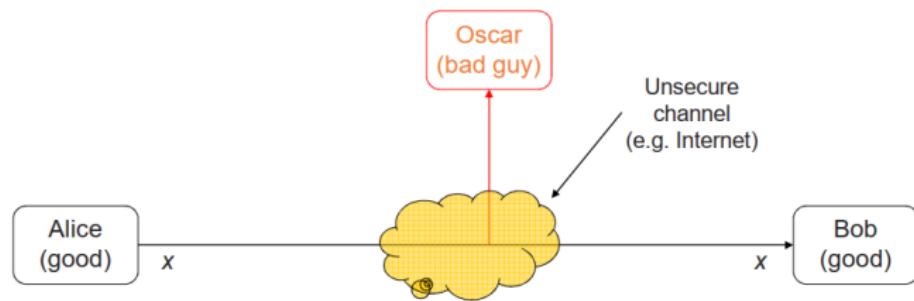
- **Criptografia Antiga:** Há indícios de criptografia no Egito por volta de 2000 a.C.  
Esquemas de substituição por letras (ex.: cifra de César) populares desde então.
- **Cifras Simétricas:** Todos os esquemas criptográficos desde a antiguidade até 1976 eram simétricos.
- **Cifras Assimétricas:** Em 1976, a criptografia de chave pública (ou assimétrica) foi proposta abertamente por Diffie, Hellman e Merkle.
- **Esquemas Híbridos:** A maioria dos protocolos atuais são esquemas híbridos, ou seja, utilizam ambos:
  - Cifras simétricas (ex.: para encriptação e autenticação de mensagens)
  - Cifras assimétricas (ex.: para troca de chaves e assinatura digital)

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Criptografia Simétrica

- Nomes alternativos: **private-key**, **single-key** ou **secret-key** cryptography.



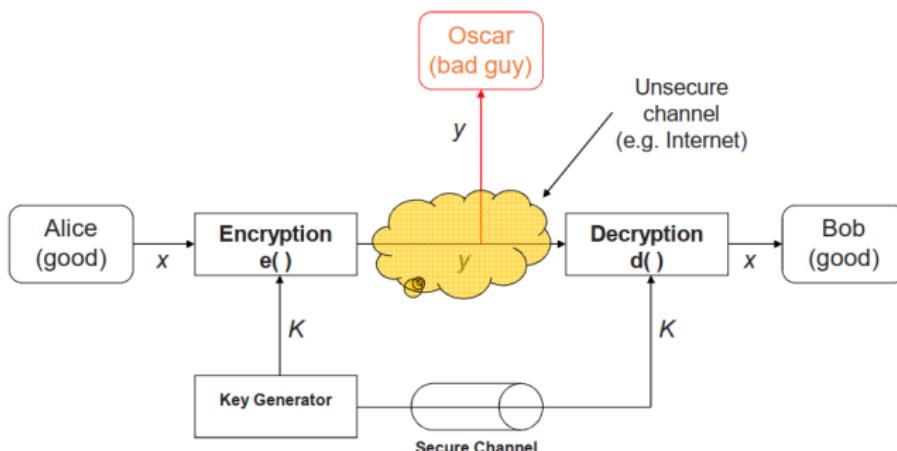
- Problema:**

- 1 Alice e Bob desejam se comunicar por um canal inseguro (ex.: WLAN ou Internet).
- 2 Um terceiro malicioso, Oscar (o "cara mau"), tem acesso ao canal, mas não deve ser capaz de entender a comunicação.

# Criptografia Simétrica

**Solução:** Criptografia com cifra simétrica.

⇒ Oscar obtém apenas o *ciphertext*  $y$ , que se parece com uma sequência aleatória de bits.



- $x$  é o **plaintext**
- $y$  é o **ciphertext**
- $K$  é a **key**
- O conjunto de todas as chaves  $\{K_1, K_2, \dots, K_n\}$  é o **key space**

# Criptografia Simétrica

## Equações

- **Equação de encriptação:**  $y = e_K(x)$
  - **Equação de desencriptação:**  $x = d_K(y)$
  - Encriptação e desencriptação são operações inversas se a mesma chave  $K$  for usada em ambos os lados:
$$d_K(y) = d_K(e_K(x)) = x$$
  - **Importante:** a chave deve ser transmitida por um **canal seguro** entre Alice e Bob.
  - Esse canal seguro pode ser realizado, por exemplo, com a instalação manual da chave (como no protocolo Wi-Fi WPA) ou via um mensageiro de confiança.
  - O sistema só é seguro se um atacante não obtiver conhecimento da chave  $K$ !
- ⇒ O problema da comunicação segura se reduz à transmissão e armazenamento seguro da chave  $K$ .

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 **Criptoanálise**
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Por que precisamos da Criptoanálise?

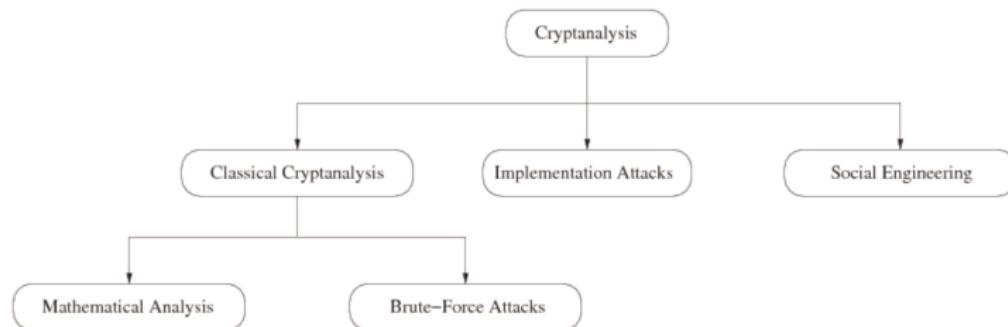
- Não existe nenhuma *mathematical proof of security* para qualquer cifra prática.
- A única forma de garantir que uma cifra é segura é tentar quebrá-la (e falhar)!

O Princípio de Kerckhoff é fundamental na criptografia moderna:

*Um sistema criptográfico deve ser seguro mesmo que o atacante (Oscar) conheça todos os detalhes sobre o sistema, com exceção da chave secreta.*

- Para atingir o Princípio de Kerckhoff na prática:  
**Utilize apenas cifras amplamente conhecidas que tenham sido criptoanalisadas por anos por bons criptógrafos!**
- **Observação:** Pode parecer que uma cifra é “mais segura” se seus detalhes forem mantidos secretos. Porém, a história mostra que cifras secretas quase sempre foram quebradas uma vez que foram revertidas por engenharia reversa.  
(Exemplo: CSS – Content Scrambling System – usado para proteção de conteúdo em DVDs.)

# Criptoanálise: Atacando Sistemas Criptográficos



- **Ataques Clássicos**
  - Análise Matemática
  - Ataque de Força Bruta
- **Ataques de Implementação:** Tentam extrair a chave por engenharia reversa ou medição de consumo de energia, por exemplo, em cartões inteligentes bancários.
- **Engenharia Social:** Por exemplo, enganar um usuário para que ele revele sua senha.

# Brute-Force Attack contra Cifras Simétricas

- Trata a cifra como uma *black box*.
- Requer (ao menos) um par *plaintext-ciphertext* ( $x_0, y_0$ ).
- Verifica todas as chaves possíveis até que a condição seja satisfeita:

$$d_K(y_0) \stackrel{?}{=} x_0$$

- Quantas chaves precisamos considerar?

Key length (bits)	Key space	Security life time
64	$2^{64}$	<b>Obsoleto / não aprovado pelo NIST</b> (força menor que 112 bits)
128	$2^{128}$	<b>Longo prazo (clássico)</b> ; planejar migração para <b>PQC</b> ; contra quântico oferece $\sim 64$ bits (insuficiente)
256	$2^{256}$	<b>Longo prazo com ampla margem (clássico)</b> ; contra quântico $\sim 128$ bits (mitiga Grover); recomendado para alvos de muito alto valor e/ou alinhamento a CNSA 2.0

**Importante:** Um atacante precisa ter sucesso em apenas **um** ataque. Logo, um grande espaço de chaves não ajuda se outros ataques (como *engenharia social*) forem possíveis.

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Cifras de Substituição

- Cifra histórica
- Excelente ferramenta para entender brute-force vs. ataques analíticos
- Cifra letras em vez de bits (como todas as cifras até o fim da Segunda Guerra Mundial)

**Ideia:** substituir cada letra do *plaintext* por outra letra fixa.

<b>Plaintext</b>	→	<b>Ciphertext</b>
A	→	k
B	→	d
C	→	w
	...	

Por exemplo, ABBA seria encriptado como kddk.

- **Exemplo (ciphertext):**

iq ifcc vqqr fb rdq vfllcq na rdq cfjwhwz hr bnnb hcc  
hwwhbsqvqbret hwq vhlq

- Quão segura é a cifra de substituição? Vamos analisar os ataques...

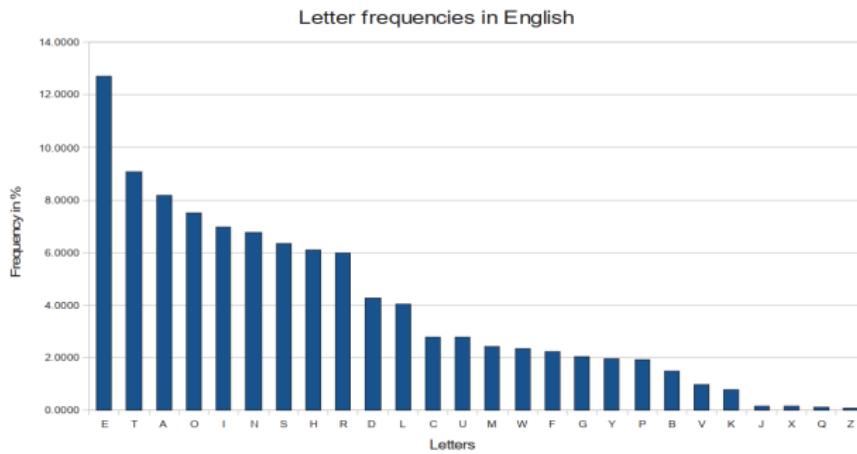
# Ataques contra a Cifra de Substituição

## ➊ Ataque: Busca Exaustiva por Chave (Brute-Force Attack)

- Basta tentar todas as tabelas de substituição possíveis até que um *plaintext* inteligível apareça.  
(Note que cada tabela de substituição é uma chave.)
- Quantas tabelas de substituição existem (= chaves)?  
$$26 \times 25 \times \cdots \times 3 \times 2 \times 1 = 26! \approx 2^{88}$$
- **Buscar em  $2^{88}$  chaves é inviável com os computadores atuais!**
- **Q:** Podemos então concluir que a *substitution cipher* é segura já que um ataque de força bruta não é viável?
- **A:** Não! Precisamos nos proteger contra **todos** os tipos de ataques...

## ② Ataque: Análise de Frequência de Letras

- As letras têm frequências muito diferentes na maioria das línguas.
- Além disso, a frequência das letras do *plaintext* é preservada no *ciphertext*.
- Por exemplo, e é a letra mais comum em inglês — quase 13% de todas as letras em um texto típico em inglês são e.
- A próxima letra mais comum é t, com cerca de 9%.



# Aplicando a Análise de Frequência de Letras

- Vamos retornar ao nosso exemplo e identificar a letra mais frequente:

i **q** ifcc v**qqr** fb rd**q** vfllc**q** na rd**q** cfjwhwz hr bnnb hcc  
hwwhbs**qvqb**re hw**q** vhl**q**

- Substituímos a letra q do *ciphertext* por E e obtemos:

i **E** ifcc v**EEr** fb rd**E** vfllc**E** na rd**E** cfjwhwz hr bnnb hcc  
hwwhbs**EvEb**re hw**E** vhl**E**

- Com mais substituições baseadas na frequência das letras restantes, obtemos o *plaintext*:

WE WILL MEET IN THE MIDDLE OF THE LIBRARY AT NOON ALL  
ARRANGEMENTS ARE MADE

# Aplicando a Análise de Frequência de Letras

- Na prática, não apenas as frequências de letras individuais podem ser usadas para ataques, mas também a frequência de pares de letras (ex.: *th* é muito comum em inglês), trios de letras, etc.

**Lição importante:** Mesmo que a **cifra de substituição** tenha um espaço de chaves grande (aproximadamente  $2^{88}$ ), ela pode ser facilmente quebrada por métodos analíticos.

Este é um excelente exemplo de que um esquema de encriptação precisa ser resistente a **todos os tipos de ataques**.

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Breve Introdução à Aritmética Modular

## Por que precisamos estudar aritmética modular?

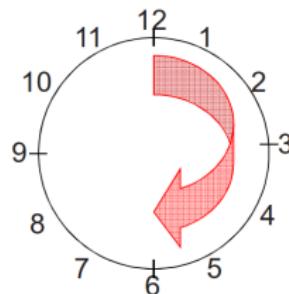
- Extremamente importante para criptografia assimétrica (RSA, curvas elípticas, etc.)
- Algumas cifras históricas podem ser elegantemente descritas com aritmética modular (veremos a cifra de César mais adiante).

# Breve Introdução à Aritmética Modular

De modo geral, a maioria dos sistemas criptográficos é baseada em **conjuntos de números** que são:

- ① **Discretos** (conjuntos com inteiros são particularmente úteis)
- ② **Finitos** (isto é, trabalhamos apenas com uma quantidade finita de números)

Parece abstrato demais? — Vamos observar um conjunto finito com números discretos com o qual estamos bem familiarizados: **um relógio**.



Curiosamente, mesmo que os números aumentem a cada hora, nunca saímos do conjunto dos inteiros:

$$1, 2, 3, \dots, 11, 12, 1, 2, 3, \dots$$

# Breve Introdução à Aritmética Modular

- Desenvolvemos agora um sistema aritmético que nos permite **calcular** em conjuntos finitos de inteiros, como os 12 números de um relógio.
- É fundamental ter uma operação que **mantém os números dentro de limites**, ou seja, após adição ou multiplicação, eles nunca devem sair do conjunto (ex.: nunca maiores que 12).

## Definição: Operação Módulo

Sejam  $a, r, m$  inteiros com  $m > 0$ . Escrevemos:

$$a \equiv r \pmod{m}$$

se  $(r - a)$  for divisível por  $m$ .

- $m$  é chamado de **módulo**
- $r$  é chamado de **resto**

## Exemplos de redução modular:

- Seja  $a = 12$  e  $m = 9$ :  $12 \equiv 3 \pmod{9}$
- Seja  $a = 37$  e  $m = 9$ :  $37 \equiv 1 \pmod{9}$
- Seja  $a = -7$  e  $m = 9$ :  $-7 \equiv 2 \pmod{9}$

(Você deve verificar se a condição “ $m$  divide  $(r - a)$ ” vale nos três casos.)

# Propriedades da Aritmética Modular (1)

- **O resto não é único**

É um pouco surpreendente, mas para um dado módulo  $m$  e número  $a$ , existem (infinitamente) muitos restos válidos.

## Exemplo:

- $12 \equiv 3 \pmod{9} \rightarrow 3$  é um resto válido pois  $9 \mid (3 - 12)$
- $12 \equiv 21 \pmod{9} \rightarrow 21$  é um resto válido pois  $9 \mid (21 - 12)$
- $12 \equiv -6 \pmod{9} \rightarrow -6$  é um resto válido pois  $9 \mid (-6 - 12)$

## Propriedades da Aritmética Modular (2)

- Qual resto devemos escolher?

Por convenção, geralmente usamos o **menor número inteiro positivo**  $r$  como o resto. Esse valor pode ser calculado como:

$$a = q \cdot m + r \quad \text{onde} \quad 0 \leq r \leq m - 1$$

**Exemplo:** Seja  $a = 12$  e  $m = 9$ :

$$12 = 1 \times 9 + 3 \quad \Rightarrow \quad r = 3$$

**Observação:** Essa escolha é apenas uma convenção.

Algorítmicamente, somos livres para usar qualquer outro resto válido ao calcular funções criptográficas.

## Propriedades da Aritmética Modular (3)

- **Como realizamos divisão modular?**

Em vez de realizar uma divisão direta, preferimos multiplicar pelo inverso modular. Por exemplo:

$$b/a \equiv b \cdot a^{-1} \pmod{m}$$

O inverso  $a^{-1}$  de um número  $a$  é definido como o número tal que:

$$a \cdot a^{-1} \equiv 1 \pmod{m}$$

**Exemplo:** Quanto é  $5/7 \pmod{9}$ ?

O inverso de 7 mod 9 é 4, pois  $7 \cdot 4 = 28 \equiv 1 \pmod{9}$ , portanto:

$$5/7 \equiv 5 \cdot 4 = 20 \equiv 2 \pmod{9}$$

# Propriedades da Aritmética Modular (4)

- **Como se calcula o inverso?**

O inverso de um número  $a \text{ mod } m$  existe se, e somente se:

$$\gcd(a, m) = 1$$

(No exemplo acima,  $\gcd(5, 9) = 1$ , então o inverso de 5 existe módulo 9.)

Atualmente, a melhor forma de calcular o inverso é por **busca exaustiva**. No futuro veremos o poderoso **Algoritmo de Euclides**, que permite encontrar o inverso modular dado um número e o módulo.

# Propriedades da Aritmética Modular (5)

- **A redução modular pode ser feita em qualquer etapa de um cálculo**

Vamos ver um exemplo. Queremos calcular  $3^8 \text{ mod } 7$  (note que exponenciação é extremamente importante em criptografia de chave pública).

## 1. Abordagem: Exponenciação seguida da redução modular

$$3^8 = 6561 \equiv 2 \pmod{7}$$

Note que o resultado intermediário foi 6561, embora o resultado final não possa ser maior que 6.

# Propriedades da Aritmética Modular (6)

## 2. Abordagem: Exponenciação com redução modular intermediária

$$3^8 = 3^4 \cdot 3^4 = 81 \cdot 81$$

Neste ponto, reduzimos  $81 \pmod{7}$ :

$$3^8 = 81 \cdot 81 \equiv 4 \cdot 4 \pmod{7}$$

$$4 \cdot 4 = 16 \equiv 2 \pmod{7}$$

Note que podemos fazer todas essas multiplicações sem calculadora, enquanto calcular mentalmente  $3^8 = 6561$  é bem desafiador para a maioria de nós.

### Regra geral

Para a maioria dos algoritmos, é vantajoso reduzir os resultados intermediários o mais cedo possível.

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Cifra de Deslocamento (ou de César) (1)

- Cifra antiga, supostamente usada por Júlio César.
- Substitui cada letra do *plaintext* por outra.
- Regra de substituição é muito simples: pega a letra que está  $k$  posições adiante no alfabeto.

É necessário mapear letras → números:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

**Exemplo para  $k = 7$ :**

- Plaintext = ATTACK = 0, 19, 19, 0, 2, 10
- Ciphertext = haaahr = 7, 0, 0, 7, 9, 17

Note que as letras “dão a volta” no final do alfabeto, o que pode ser descrito matematicamente como redução módulo 26.

Exemplo:  $19 + 7 = 26 \equiv 0 \pmod{26}$

## Cifra de Deslocamento (ou de César) (2)

- Descrição matemática elegante da cifra:

Sejam  $k, x, y \in \{0, 1, \dots, 25\}$

- **Encriptação:**  $y = e_k(x) \equiv x + k \pmod{26}$
- **Decriptação:**  $x = d_k(y) \equiv y - k \pmod{26}$

- **Pergunta:** A cifra de César é segura?
- **Resposta:** Não! Vários ataques são possíveis, incluindo:
  - Busca exaustiva (espaço de chaves tem apenas 26 possibilidades!)
  - Análise de frequência de letras (como na cifra de substituição)

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Cifra Afim (Affine) (1)

- Extensão da cifra de deslocamento: em vez de apenas somar a chave ao texto-claro, também multiplicamos por ela.
- Usamos uma chave com duas partes:  $k = (a, b)$ .

Sejam  $k, x, y \in \{0, 1, \dots, 25\}$ .

- **Encriptação:**  $y = e_k(x) \equiv ax + b \pmod{26}$
- **Decriptação:**  $x = d_k(y) \equiv a^{-1}(y - b) \pmod{26}$

- Como o inverso de  $a$  é necessário para decifrar, só podemos usar valores de  $a$  tais que  $\gcd(a, 26) = 1$ .
- Existem 12 valores de  $a$  que atendem a essa condição (por exemplo,  $a \in \{1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25\}$ ).
- Disso decorre que o espaço de chaves é apenas  $12 \times 26 = 312$ .
- Novamente, vários ataques são possíveis, incluindo:
  - busca exaustiva e análise de frequência de letras, semelhante ao ataque contra a cifra de substituição.

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

# Leis de Segurança de Shamir (Prêmio Turing 2002)

- Sistemas completamente seguros não existem.
- Para diminuir suas vulnerabilidades pela metade, deve-se dobrar os gastos.
- Criptografia é normalmente contornada, não quebrada.

*Adi Shamir é coautor do algoritmo RSA e recebeu o Prêmio Turing em 2002.*



# Lições Aprendidas

- Nunca, jamais, desenvolva seu próprio algoritmo criptográfico sem ter uma equipe de criptanalistas experientes revisando seu projeto.
- Não utilize algoritmos ou protocolos criptográficos que não foram devidamente comprovados.
- Atacantes sempre buscam o ponto mais fraco de um sistema criptográfico. Um grande espaço de chaves, por si só, não garante segurança — a cifra pode ser vulnerável a ataques analíticos.
- Tamanhos de chave recomendados para algoritmos simétricos, a fim de evitar ataques por busca exaustiva:
  - **64 bits:** inseguro, exceto para dados com valor de curtíssimo prazo.
  - **128 bits:** segurança de longo prazo (várias décadas), a menos que computadores quânticos se tornem realidade (o que talvez nunca ocorra).
  - **256 bits:** como acima, mas provavelmente seguro mesmo contra ataques quânticos.
- A aritmética modular é uma ferramenta elegante para descrever esquemas históricos de encriptação, como a cifra de César, de forma matemática.

# Índice

- 1 Visão geral da Criptologia
- 2 Conceitos Básicos de Criptografia Simétrica
- 3 Criptoanálise
- 4 Cifras de Substituição
- 5 Aritmética Modular
- 6 Cifra de César (ou Deslocamento)
- 7 Cifra Afim (Affine)
- 8 Conclusão
- 9 Exercícios

## Exercício: Segurança de Longo Prazo do AES

Considere a segurança de longo prazo do AES com chave de 128 bits diante de ataques por busca exaustiva (força bruta).

- ➊ Suponha que um atacante possua chips dedicados (*ASICs*) capazes de testar  $5 \cdot 10^8$  chaves por segundo. O orçamento total é de \$1 milhão. Cada *ASIC* custa \$50, e assumimos 100% de custo adicional para integração (placa, alimentação, refrigeração etc).
  - Quantos *ASICs* podem ser utilizados em paralelo?
  - Qual o tempo médio para encontrar a chave correta?
  - Compare esse tempo com a idade do Universo ( $\approx 10^{10}$  anos).
- ➋ Agora, considere avanços tecnológicos. A Lei de Moore prevê que o poder computacional dobra a cada 18 meses, enquanto o custo dos circuitos integrados permanece constante.
  - Quantos anos teremos que esperar até que uma máquina de busca exaustiva possa quebrar o AES-128 em, no máximo, 24 horas? Assuma novamente um orçamento de \$1 milhão (desconsidere inflação).

# Solução (Parte 1/2): Capacidade Atual de Ataque

## Dados:

- Orçamento: \$1 milhão
- Custo por ASIC (com overhead): \$100
- $\Rightarrow$  Número de ASICs:  $1\,000\,000 \div 100 = 10\,000$
- Velocidade de cada ASIC:  $5 \times 10^8$  chaves/s

## Capacidade total:

$$10\,000 \times 5 \times 10^8 = 5 \times 10^{12} \text{ chaves/s}$$

## Tempo médio para encontrar a chave correta:

$$T = \frac{2^{127}}{5 \times 10^{12}} \approx 4.25 \times 10^{25} \text{ segundos}$$

$$\Rightarrow \text{em anos: } \frac{4.25 \times 10^{25}}{60 \times 60 \times 24 \times 365} \approx 1.35 \times 10^{18} \text{ anos}$$

**Conclusão:** Muito maior que a idade do Universo ( $\sim 10^{10}$  anos).

## Solução (Parte 2/2): Projeção com a Lei de Moore

### Objetivo:

- Reduzir o tempo médio de quebra para no máximo 24 horas (86400 segundos).

### Cálculo:

$$\frac{2^{127}}{V} \leq 86400 \quad \Rightarrow \quad V \geq \frac{2^{127}}{86400}$$
$$V \approx 1.46 \times 10^{34} \text{ chaves/s}$$

### Comparação com a capacidade atual:

Velocidade atual:  $5 \times 10^{12}$  chaves/s

Número de duplicações necessárias:  $\log_2 \left( \frac{1.46 \times 10^{34}}{5 \times 10^{12}} \right) \approx 74$

### Tempo estimado considerando a Lei de Moore:

$$74 \times 1.5 = 111 \text{ anos}$$

## Exercício: Relação entre Senhas e Tamanho da Chave

Consideramos agora a relação entre senhas e o tamanho da chave. Para isso, analisamos um sistema criptográfico no qual o usuário insere uma chave na forma de uma senha.

- ① Suponha uma senha composta por 8 caracteres, onde cada caractere é codificado com ASCII (7 bits por caractere, ou seja, 128 possíveis caracteres).

Qual é o tamanho do espaço de chaves que pode ser construído com essas senhas?

- ② Qual é o comprimento correspondente da chave em bits?
- ③ Suponha que a maioria dos usuários utilize apenas as 26 letras minúsculas do alfabeto, em vez dos 128 caracteres ASCII.  
Qual seria o comprimento correspondente da chave em bits nesse caso?
- ④ Quantos caracteres são necessários, no mínimo, para gerar uma chave de 128 bits no caso de:
  - ① Caracteres de 7 bits?
  - ② Letras minúsculas do alfabeto (26 símbolos)?

# Solução: Relação entre Senhas e Tamanho da Chave

## ① Espaço de chaves com 8 caracteres ASCII (7 bits):

Cada caractere:  $2^7 = 128$  possibilidades.

Total para 8 caracteres:  $128^8 = 2^{56}$  combinações.

## ② Tamanho da chave em bits:

$2^{56}$  combinações correspondem a uma chave de **56 bits**.

## ③ Se o usuário usa apenas 26 letras minúsculas:

Espaço de chaves:  $26^8$  combinações.

$26^8 \approx 2^{37.6}$ , logo a chave tem aproximadamente **38 bits**.

## ④ Mínimo de caracteres necessários para atingir 128 bits de segurança:

### ① Para caracteres de 7 bits:

$$\text{Precisamos de } 2^{7n} \geq 2^{128} \Rightarrow n \geq \frac{128}{7} \approx 18.3 \Rightarrow \mathbf{19 \text{ caracteres}}$$

### ② Para letras minúsculas (26 símbolos):

$$26^n \geq 2^{128} \Rightarrow n \geq \frac{128}{\log_2 26} \approx \frac{128}{4.7} \approx \mathbf{27.2 \Rightarrow 28 \text{ caracteres}}$$

## Exercício: Cifra de Vigenère

Consideramos uma extensão da cifra de César: a cifra de **Vigenère**, que utiliza uma sequência de deslocamentos  $k_i$  derivada de uma palavra-código  $c = (c_0, c_1, \dots, c_{l-1})$  com  $l$  letras.

Cada letra  $c_i$  é mapeada para um número de 0 a 25, baseado em sua posição no alfabeto ( $A = 0, B = 1, \dots, Z = 25$ ). A cifra aplica deslocamentos cíclicos:

$$y_j \equiv x_j + k_j \bmod l \quad \bmod 26$$

1. Dada a palavra-código JAMAIKA ( $l = 7$ ), converta as letras nos valores  $k_i$  correspondentes.
2. Utilize a tabela de substituição (alfabeto deslocado) para cifrar a palavra CODEBREAKERS.

Para cada letra, aplique o deslocamento  $k_j \bmod l$  correspondente.

3. Analise a segurança da cifra de Vigenère.

Você consegue imaginar um ataque contra ela? Como descobrir a palavra-código?

# Tabela de Deslocamento para Cifra Polialfabética

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y